Client's ref.: VIT03-0095
File:0608-10297US /Karen

TITLE

METHOD AND SYSTEM OF SIGNAL PATH TRACKING

BACKGROUND OF THE INVENTION

Field of the Invention

The present invention relates to symbol timing recovery, and more specifically, to a method and system of signal path tracking to recover synchronization between a receiver and a received signal.

Description of the Related Art

In digital receiving, sampling examines the received signal at specific times for symbol detection. The sampling process samples the received signal at a preset sampling rate, and each symbol may have more than one sample. The sampling process also includes establishing an on-time sample for each symbol. These on-time samples will be used to identify the symbols carried in the received signal. The on-time samples are best established when the signal has a highest signal to noise ratio (SNR) in every symbol duration. A match filter is usually included in the digital receiving system, in order to generate a peak signal strength which helps suppress effects of noise. The receiver needs to be synchronized with the receiving signal, and makes on-time samples at the optimal points. However, since the accuracy and stability of the clock reference in the receiver are not ideal, and the condition of transmission channel may vary significantly, this synchronization needs to be maintained. Sampling at the optimal points is very important because even a small offset in sampling point may cause significant errors.

Thus, the receiver requires a symbol timing recovery block to keep track of the received signal and determine the optimal points for symbol sampling. The symbol timing recovery block maintains synchronization by tracking the signal path and periodically adjusting the timing scheme for establishing on-time samples.

Figure 1 is a block diagram showing a digital receiving system as disclosed. The receiver receives a signal by an antenna 102. A demodulator 104 receives the signal from the antenna 102, and demodulates the signal using a frequency identical to that used in the transmitter. The transmission channel may introduce a phase offset into the signal, and for that reason, a carrier recovery block 106 is usually required to remove the phase offset. This carrier recovery block 106 is usually implemented by a phase lock loop (PLL). A match filter 108 receives the signal from the demodulator 104. The match filter 108 is designed to match the shape of the signal, in order to generate a peak SNR for each symbol. The symbol sampler 110 receives and samples the matched signal. The symbol timing recovery block 112 checks and controls the sampling points. Data from the symbol sampler 110 is then recovered into digital data through quantization and un-mapping in block 114.

The conventional symbol timing recovery block 112 is usually a delay lock loop (DLL). Figure 2 shows a simple block diagram for the delay lock loop with the symbol sampler. The symbol sampler 202 takes in samples at a preset rate, and the system also decides the number of samples per symbol. The symbol sampler 202 establishes an on-time sample according to a timing scheme. The DLL then checks the validity of the on-time sample, subtracting the previous sample (early sampling) 206 from the

subsequent sample (late sampling) 208. Ideally, if the on-time sample peaks the match filter output, the early and late sampling are equal. Since this is rarely the case, differences between the early and late samplings are respectively multiplied by the on-time sampling value and added. When this sum exceeds a threshold, it is clear that the symbol sampler 202 is not sampling at an allowable duration for correct symbol detection. The offset decision 204 then adjusts the timing scheme to generate a more precise sampling point for the next symbol. This DLL system thus fixes the optimal points for sampling after a few trials.

Although the DDL system can recover the sampling points when tracking a single signal, the DDL system becomes less reliable and may not perform symbol timing recovery properly in the environment of clustered signal paths, since the early sampling and the late sampling will be seriously affected by the paths of other signals.

SUMMARY OF THE INVENTION

The object of the present invention is to provide a method of signal path tracking for symbol timing recovery.

Another object of the present invention is to provide a method for symbol timing recovery which can track signal path in an environment of clustered signal paths.

In order to achieve these objects, the present invention provides a signal path tracking method and system for symbol timing recovery. The signal path tracking system includes a symbol sampler, which samples a received signal according to a timing scheme. Current sampling points are expected to establish on-time samples for current symbols. The symbol

sampler samples at a preset sampling rate with a fixed number of samples per symbol. A peak detector receives the samples from the symbol sampler and outputs optimal points for establishing on-time samples of the current symbols. The peak detector determines the optimal point for each symbol by finding a sample with strongest SNR during the duration of each symbol, and outputs the corresponding time of the sample as the optimal point.

The current sampling points of the symbol sampler and the optimal points detected by the peak detector are input to an error detector. The error detector computes an expected error by calculating timing differences between the two inputs, filtering the timing differences to obtain an average moving error, and finally integrating the average moving error. The error detector outputs the expected error to a path tracker to predict a future sampling point of a subsequent symbol. The path tracker then feeds back the future sampling point to the symbol sampler to adjust the timing scheme, thereby recovering symbol timing of the receiver.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention can be more fully understood by reading the subsequent detailed description in conjunction with the examples and references made to the accompanying drawings, wherein:

Fig. 1 is a block diagram showing a digital receiving system;

Fig. 2 is a block diagram showing a delayed locked loop (DLL) for symbol timing recovery;

Fig. 3 is a block diagram showing a first embodiment of the signal path tracking system of the present invention;

Fig. 4 is a signal-time graph showing the sampled signal;

Fig. 5A is a simple block diagram of the error detector in Fig. 3;

Fig. 5B is a logic circuit diagram of the error detector;

Fig. 6 is a block diagram showing a second embodiment of the signal path tracking system of the present invention;

Fig. 7 is a flowchart showing the signal path tracking method of the second embodiment;

Fig. 8 is a graph illustrating an example of the signal path tracking according to the second embodiment;

Fig. 9A is a graph showing the simulation result of a static signal;

Fig. 9B is a graph showing the simulation result of a dynamic signal;

Fig. 10A is a graph showing the average moving error of tracking the static signal in Fig. 9A;

Fig. 10B is a graph showing the average moving error of tracking the dynamic signal in Fig. 9B;

Fig. 11 is a graph showing the simulation result of two dynamic signals;

Fig. 12 is a graph showing the average moving error of tracking the two dynamic signals in Fig. 11.

DETAILED DESCRIPTION OF THE INVENTION

Figure 3 is a block diagram showing an embodiment of the signal path tracking system disclosed in the present invention. The signal path tracking system is an alternative for the conventional delay lock loop (DLL). The purpose of the signal

path tracking system 30 is to adjust the timing scheme of the symbol sampler 302, in order to establish on-time samples at the optimal points. The symbol sampler 302 samples the output of a match filter 300 at a preset sampling rate. This sampling rate takes a desired number of samples per symbol according to the frequency of the received signal. In Figure 4, the signal-time diagram shows a signal sampled at a sampling rate of 8 samples per symbol. As mentioned before, the match filter 300 generates a peak signal during each symbol duration, the optimal point. The symbol sampler establishes an on-time sample for each symbol according to the timing scheme, with the on-time sample examined to determine the original symbol in the later stages of the receiving method, such that the on-time samples must be precise enough to recover the original symbols. Taking the on-time samples at optimal points is an effective approach to increase accuracy.

First embodiment

As shown in Figure 3, a peak detector 304 determines the time of the peak signal in each symbol, creating optimal point 314, received by an error detector 306, along with current sampling point 310 from a path tracker 308. The symbol sampler 302 updates its timing scheme according to an output 318 of the path tracker 308. Details of these functional blocks are described in the following.

Current sampling points 402a~402d of Figure 4 correspond to points at which symbol sampler 302 of Figure 3 samples the on-time sample for each symbol. In Figure 4, the current sampling points 402a~402d mark acquisition of the fifth samples of the symbols. After obtaining an on-time sample, the symbol

sampler 302 simply takes another 8 samples to establish the next on-time sample for a subsequent symbol.

Due to channel imperfection, interference from other signals, or other possible causes, the timing scheme of the receiver has difficulty synchronizing with the received signal after a period of time. Thus the receiver must perform signal path tracking to adjust the timing scheme to recover the synchronization periodically. The example given in Figure 4 shows that although peak amplitudes are set to be sampled at current sampling points 402a~402d, actual peaks occur at 404a~404d, points not always aligned with the current sampling points 402a~402d, defined as optimal points. In Figure 4, the optimal points 404c and 404d correspond with the fourth samples, different from the current sampling points 402c and 402d. In this example, the peak value of each symbol gradually shifts from the fifth sample to the fourth sample, which implies that timing of the receiver may be required to align with the receiving signal. A symbol timing recovery system is required to modify the timing scheme periodically to synchronize the optimal points for subsequent symbols.

A peak detector 304 in Figure 3 receives samples from the symbol sampler 302, and then detects the time of peak signal strength (optimal points 314) in each symbol duration. An error detector 306 error checks once every N symbols. The error detector 306 receives N optimal points 314 from the peak detector 304 and N current sampling point 310 from the path tracker 308.

Figure 5A is a simple block diagram showing three functional blocks in the error detector 306. Figure 5B is a more detailed diagram showing logic circuits of the three functional blocks in Figure 5A. If N symbols are located in a slot and the

error detector computes an output when receiving a complete slot of symbols, when the error detector processes slot k, it must receive N current sampling points $x_{real}(n)$ 310 generated according to the current timing scheme, used for sampling symbols in slot k. The error detector also receives N optimal points $x_{opt}(n)$ 314 found by detecting the peak value in each symbol of slot k, where n=0~N. First, an average sampling trace XR_k is determined by averaging N x_{real} in slot k, and an average optimal trace XO_k is also determined in a similar manner, as shown in equations (1) and (2).

$$XR_k = \sum_{n=1}^{N} x_{real}(n)/N \qquad \text{-------- (1)}$$

$$XO_k = \sum_{n=1}^{N} x_{opt}(n)/N \qquad \text{--------- (2)}$$

These average operations are performed by the two average circuits 501 and 503. XR_k 506 and XO_k 508 then feed to a comparator 500 to execute the following equation.

$$E_k = XR_k - XO_k \qquad \text{--------- (3)}$$

The difference between the two inputs 506 and 508 is an error E_k 510, which indicates the validity of current on-time sampling. The signal path tracking system seeks to minimize this error 510 for the next slot. A loop filter 502 receives E_k 510 and the loop filter 502 stores at least one error from an earlier slot, for example, E_{k-1}. By extrapolation algorithm or other similar mathematical estimation, E_k and E_{k-1} can be used to predict an expected error EE_{k+1} at the end of slot k+1, or an average moving error ee_{k+1} contributed per symbol in slot k+1. Such extrapolation can be performed by convolution in the loop filter 502 as shown in equation (4),

$$ee_{k+1} = EE_{k+1}/N = (E_k, E_{k-1}, E_{k-2}, \ldots) \otimes F(s) \qquad \text{------- (4)}$$

where F(s) represents a loop filter with coefficients. As shown in Figure 5B, the loop filter 502 has two convolution operators and a delay register which accumulates precedent values and computes the average moving error ee_{k+1} 512. ee_{k+1} 512 indicates an average rate of change of the input 510. This expected error EE_{k+1} or average moving error ee_{k+1} should be used to adjust the current timing scheme, in order to minimize E_{k+1} in the next slot (slot k+1). In the next slot, $x_{real}(n)$ calculated from the adjusted timing scheme will hence be $n*ee_{k+1}$ ahead of the $x_{real}(n)$ calculated from the unadjusted time scheme after adjustment.

$$x_{real}(n) = init\_addr - \int ee_{k+1} dt = init\_addr - n * EE_{k+1} / N \qquad \text{------ (5)}$$

init_addr in equation (5) represents $x_{real}(n)$ calculated from the unadjusted timing scheme for slot k+1. In Figure 5A, the loop filter 502 passes the average moving error 512 to an integrator 504. The integrator 504 computes expected errors 316 according to the average moving error 512 to correct future sampling points for the next N symbols.

Still in Figure 3, the expected errors 316 for estimating the sampling points of the next N symbols discussed before are the input of a path tracker 308. The path tracker 308 computes the current sampling points 310 and 318 according to the expected errors 316. The symbol sampler 302 receives the sampling points 318 as feedback for symbol timing recovery. This feedback action adjusts the timing scheme of the symbol sampler 302, thereby recovering the synchronization between the receiver and the receiving signal.

Second embodiment

Figure 6 is a block diagram illustrating a second embodiment of the signal path tracking system of the present

invention. The second embodiment can be implemented in a RAKE receiver for the CDMA system. Instead of a single signal, a cluster of signals is received by the match filter 300 of the receiver. The match filter 300 then inputs the matched signals into the symbol sampler 302. The match filter 300 and the symbol sampler 302 of the second embodiment are identical to the first embodiment. The output of the symbol sampler 302 is connected to an input of a path searcher 602. The path searcher 602 identifies the sampled signals, and forwards the sampled signals to a path assignment 604. The path assignment acts like a multiplexer, which separates the sampled signals, then forwards each of the predetermined k sets of signal path to a corresponding moving error detector 606 (referred as 'finger' in a RAKE receiver). Figure 6 shows a case of receiving four independent signal paths (K=4), hence four moving error detectors 606a~606d and four path moving trackers 608a~608d are used. Figure 7 is a flowchart describing the process of the moving error detector 606 and the path moving tracker 608. Each of the moving error detector 606 averages N data of the signal path received from the path assignment 604 to obtain an averaged path Av_addr. The moving error detector 606 then calculates an average moving error E_{error} by filtering the difference e_{error} between the averaged path Av_addr and an estimated path IP_addr. Each of the moving error detector 606 passes the average moving error E_{error} and an initial path Init_Add0 to a corresponding path moving tracker 608. Each of the path moving tracker 608 integrates the average moving error E_{error} to obtain an integrated error IE_{error}, and modifies the initial path Init_Add0 according to the integrated error IE_{error} to obtain the estimated path IP_addr.

Figure 8 is a graph illustrating the signal path tracking according to the second embodiment of the present invention. The horizontal axis of the graph represents time. The estimated path IP_addr is computed to track the signal path X_{real}. The averaged path Av_addr is shown as horizontal lines in each slot, and the difference between the averaged path Av_addr and the estimated path IP_addr is e_{error}. However, there is still an offset X_{offset} between the signal path X_{real} and the estimated path IP_addr. This offset could be reduced by increasing the complexity of the hardware and the processing speed of the central processor.

Figures 9~12 are simulation results of signal path tracking with a chip rate of 3.84 MHz, and a sampling rate of four times the chip rate (15.36 MHz). In Figure 9A, curve (1) represents a static signal after sampling, curve (2) represents the estimated path (IP_addr) of the static signal computed by the signal path tracking method of the present invention, and curve (3) represents the real path (X_{real}) of the static signal. In Figure 9B, curve (1) represents a dynamic signal after sampling, curve (2) represents the estimated path (IP_addr) of the dynamic signal computed by the signal path tracking method of the present invention, and curve (3) represents the real real path (X_{real}) of the dynamic signal. The dynamic signal has an average moving error of 7ppm. The curve in Figure 10A shows the average moving error of the static signal in Figure 9A. The curve in Figure 10B shows the average moving error of the dynamic signal in Figure 9B.

Simulation results in Figures 11 and 12 prove that the signal path tracking method of the present invention is also capable of tracking the path of a cluster of signals. In Figure 11, curve (1) represents the path of two dynamic signals (path

1, path 2), curve (2) represents the estimated path (IP_addr) of the two dynamic signals computed by the signal path tracking method of the present invention, and curve (3) represents the real path (X_{real}) of the two dynamic signals (path 1, path 2). The two dynamic signals both have an average moving error of 7ppm, and the paths of the two dynamic signals are only two-sample apart. The curve in Figure 12 shows the average moving error of the clustered path.

The present invention provides a method of signal path tracking for symbol timing recovery using extrapolation, predicting outer data by two or more true measurements. In comparison with the DLL system, the symbol timing recovery system of the present invention detects the optimal points for sampling the on-time sample and compares them with the current sampling points. The system then adjusts the timing scheme for subsequent symbols according to the average moving error of the timing differences. The DLL system, on the other hand, detects timing error by comparing a sample before and after the on-time sample, then modifying the timing scheme if the difference between these two samples exceeds a preset threshold.

The method disclosed in the present invention is capable of tracking a cluster of signals in a complicated channel, for example a CDMA channel. The signal tracking system of the present invention can be implemented in a RAKE receiver for receiving CDMA signals.

Finally, while the invention has been described by way of example and in terms of the above, it is to be understood that the invention is not limited to the disclosed embodiment. On the contrary, it is intended to cover various modifications and similar arrangements as would be apparent to those skilled in

the art. Therefore, the scope of the appended claims should be accorded the broadest interpretation so as to encompass all such modifications and similar arrangements.